



COIT12200 *Software Design & Development*

Term 1 - 2020

Profile information current as at 04/05/2024 06:17 pm

All details in this unit profile for COIT12200 have been officially approved by CQUniversity and represent a learning partnership between the University and you (our student). The information will not be changed unless absolutely necessary and any change will be clearly indicated by an approved correction included in the profile.

Corrections

Unit Profile Correction added on 24-04-20

The end of term examination has now been changed to an alternate form of assessment. Please see your Moodle site for details of the assessment.

General Information

Overview

This unit provides students with practical knowledge and skills for software design and development, including the phases of requirements engineering, system modelling, architectural design, software implementation, testing and evolution in the Systems Development Lifecycle (SDLC). In this unit, you will construct layered information systems, consisting of presentation and application and data layers, to demonstrate the understanding of SDLC processes. Completion of this unit will help build a solid foundation for software application development.

Details

Career Level: *Undergraduate*

Unit Level: *Level 2*

Credit Points: 6

Student Contribution Band: 8

Fraction of Full-Time Student Load: 0.125

Pre-requisites or Co-requisites

Pre-requisites: (COIT11134 and COIT11237) OR (COIT11134 and COIT12167)

Important note: Students enrolled in a subsequent unit who failed their pre-requisite unit, should drop the subsequent unit before the census date or within 10 working days of Fail grade notification. Students who do not drop the unit in this timeframe cannot later drop the unit without academic and financial liability. See details in the [Assessment Policy and Procedure \(Higher Education Coursework\)](#).

Offerings For Term 1 - 2020

- Brisbane
- Cairns
- Melbourne
- Online
- Rockhampton
- Sydney
- Townsville

Attendance Requirements

All on-campus students are expected to attend scheduled classes – in some units, these classes are identified as a mandatory (pass/fail) component and attendance is compulsory. International students, on a student visa, must maintain a full time study load and meet both attendance and academic progress requirements in each study period (satisfactory attendance for International students is defined as maintaining at least an 80% attendance record).

Website

[This unit has a website, within the Moodle system, which is available two weeks before the start of term. It is important that you visit your Moodle site throughout the term. Please visit Moodle for more information.](#)

Class and Assessment Overview

Recommended Student Time Commitment

Each 6-credit Undergraduate unit at CQUniversity requires an overall time commitment of an average of 12.5 hours of study per week, making a total of 150 hours for the unit.

Class Timetable

[Regional Campuses](#)

Bundaberg, Cairns, Emerald, Gladstone, Mackay, Rockhampton, Townsville

[Metropolitan Campuses](#)

Adelaide, Brisbane, Melbourne, Perth, Sydney

Assessment Overview

1. **Practical and Written Assessment**

Weighting: 20%

2. **Practical and Written Assessment**

Weighting: 40%

3. **Examination**

Weighting: 40%

Assessment Grading

This is a graded unit: your overall grade will be calculated from the marks or grades for each assessment task, based on the relative weightings shown in the table above. You must obtain an overall mark for the unit of at least 50%, or an overall grade of 'pass' in order to pass the unit. If any 'pass/fail' tasks are shown in the table above they must also be completed successfully ('pass' grade). You must also meet any minimum mark requirements specified for a particular assessment task, as detailed in the 'assessment task' section (note that in some instances, the minimum mark for a task may be greater than 50%). Consult the [University's Grades and Results Policy](#) for more details of interim results and final grades.

CQUniversity Policies

All University policies are available on the [CQUniversity Policy site](#).

You may wish to view these policies:

- Grades and Results Policy
- Assessment Policy and Procedure (Higher Education Coursework)
- Review of Grade Procedure
- Student Academic Integrity Policy and Procedure
- Monitoring Academic Progress (MAP) Policy and Procedure – Domestic Students
- Monitoring Academic Progress (MAP) Policy and Procedure – International Students
- Student Refund and Credit Balance Policy and Procedure
- Student Feedback – Compliments and Complaints Policy and Procedure
- Information and Communications Technology Acceptable Use Policy and Procedure

This list is not an exhaustive list of all University policies. The full list of University policies are available on the [CQUniversity Policy site](#).

Previous Student Feedback

Feedback, Recommendations and Responses

Every unit is reviewed for enhancement each year. At the most recent review, the following staff and student feedback items were identified and recommendations were made.

Feedback from Student evaluation

Feedback

Students found the theoretical concepts lengthy and exhaustive in the lecture and tutorials.

Recommendation

Include a mix of both theoretical and practical activities in the tutorials. More practical activities on UML modelling will be discussed in the tutorials.

Feedback from Student evaluation

Feedback

Students found the case study used for modelling and implementation in parallel to the assessments very helpful.

Recommendation

Continue this practice.

Feedback from Self-reflection

Feedback

Provide more detailed feedback in assessment marking.

Recommendation

Maintain communication with the teaching team about the importance of detailed comments in the assessments. These comments will enhance the students understanding and learning capacity.

Unit Learning Outcomes

On successful completion of this unit, you will be able to:

1. Apply software design and development principles
2. Complete appropriate documentation for requirements, system modelling and design, software implementation, testing and evolution
3. Illustrate a detailed design for a layered information system using appropriate design methods
4. Implement an information system, consisting of presentation, application and data layers
5. Discuss the issues involved in software evolution and maintenance
6. Use quality management in a range of different design and development scenarios.

Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA is in use in over 100 countries and provides a widely used and consistent definition of ICT skills. SFIA is increasingly being used when developing job descriptions and role profiles.

ACS members can use the tool MySFIA to build a skills profile at

<https://www.acs.org.au/professionalrecognition/mysfia-b2c.html>

This unit contributes to the following workplace skills as defined by SFIA. The SFIA code is included:

- Programming/Software Development (PROG)
- Testing (TEST).
- System design (DESN)
- System Integration (SINT)
- Quality management (QUMG)
- Requirements definition and management (REQM)
- Business modelling (BSMO)

Alignment of Learning Outcomes, Assessment and Graduate Attributes

 N/A Level	 Introductory Level	 Intermediate Level	 Graduate Level	 Professional Level	 Advanced Level
---	--	--	--	--	--

Alignment of Assessment Tasks to Learning Outcomes

Assessment Tasks	Learning Outcomes					
	1	2	3	4	5	6
1 - Practical and Written Assessment - 20%		•	•	•		
2 - Practical and Written Assessment - 40%	•	•	•	•		•
3 - Examination - 40%	•				•	•

Alignment of Graduate Attributes to Learning Outcomes

Graduate Attributes	Learning Outcomes					
	1	2	3	4	5	6
1 - Communication	•	•	•		•	•
2 - Problem Solving	•	•	•	•	•	•
3 - Critical Thinking			•	•	•	
4 - Information Literacy	•	•	•		•	•
5 - Team Work						
6 - Information Technology Competence	•	•	•	•		•
7 - Cross Cultural Competence						
8 - Ethical practice				•		
9 - Social Innovation						
10 - Aboriginal and Torres Strait Islander Cultures						

Alignment of Assessment Tasks to Graduate Attributes

Assessment Tasks	Graduate Attributes									
	1	2	3	4	5	6	7	8	9	10
1 - Practical and Written Assessment - 20%	•	•	•	•		•		•		
2 - Practical and Written Assessment - 40%	•	•	•	•		•		•		
3 - Examination - 40%	•	•	•	•		•				

Textbooks and Resources

Textbooks

COIT12200

Prescribed

Software Engineering Global Edition (10e)

Edition: 10 (2015)

Authors: Ian Sommerville

Pearson Education Limited

Harlow , United Kingdom

ISBN: 9781292096131

Binding: Paperback

Additional Textbook Information

Copies can be purchased at the CQUni Bookshop here: <http://bookshop.cqu.edu.au> (search on the Unit code)

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Java Development Kit (JDK) 1.8 or later
- The NetBeans IDE
- MySql Database Server version 5.7 or later
- Violet Uml editor 2.1.0 or later

Referencing Style

All submissions for this unit must use the referencing style: [Harvard \(author-date\)](#)

For further information, see the Assessment Tasks.

Teaching Contacts

Farzad Sanati Unit Coordinator

f.sanati@cqu.edu.au

Schedule

Week 1 - 09 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
Introduction and Software Processes	Chapters 1 and 2	

Week 2 - 16 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
Requirements engineering	Chapter 4	

Week 3 - 23 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
System modelling	Chapter 5	

Week 4 - 30 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
Architectural design	Chapter 6	

Week 5 - 06 Apr 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Design and Implementation	Chapter 7	
Vacation Week - 13 Apr 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Week 6 - 20 Apr 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Agile software development	Chapter 3	Assignment 1 Due: Week 6 Friday (24 Apr 2020) 11:45 pm AEST
Week 7 - 27 Apr 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Software testing	Chapter 8	
Week 8 - 04 May 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Software evolution and Software Reuse	Chapters 9 and 11	
Week 9 - 11 May 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Quality management	Chapter 24	
Week 10 - 18 May 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Configuration management	Chapter 25	
Week 11 - 25 May 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Dependable systems	Chapter 10	Assignment 2 Due: Week 11 Friday (29 May 2020) 11:45 pm AEST
Week 12 - 01 Jun 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Security engineering	Chapter 13	
Review/Exam Week - 08 Jun 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Exam Week - 15 Jun 2020		
Module/Topic	Chapter	Events and Submissions/Topic

Term Specific Information

Unit Coordinator: Dr. Farzad Sanati
School of Engineering and Technology Central Queensland University
Townsville QLD 3810, Australia
Phone: +61 7 4726 5386
Email: f.sanati@cqu.edu.au

Assessment Tasks

1 Assignment 1

Assessment Type

Practical and Written Assessment

Task Description

You will demonstrate your ability in applying the principles of requirement engineering by eliciting the functional and non-functional requirements, documenting requirements specification, modelling the system and designing the system architecture and implementation. In this assignment task, you will be using the topics learned in Weeks 1-5. Complete details are in the Assignment 1 Specification document available from the Unit website.

Assessment Due Date

Week 6 Friday (24 Apr 2020) 11:45 pm AEST

Return Date to Students

Week 8 Friday (8 May 2020)

Weighting

20%

Assessment Criteria

This is an individual assignment and contributes to 20% of the total marks.

- Apply software design and development principles.
- Complete appropriate documentation for requirements, system modelling, and design, software implementation.
- Illustrate a detailed design for a layered information system using appropriate design methods.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Submission Instructions

Submit a document file (.doc or .docx) using the Assignment 1 submission link available from the Unit website.

Learning Outcomes Assessed

- Complete appropriate documentation for requirements, system modelling and design, software implementation, testing and evolution
- Illustrate a detailed design for a layered information system using appropriate design methods
- Implement an information system, consisting of presentation, application and data layers

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Information Technology Competence
- Ethical practice

2 Assignment 2

Assessment Type

Practical and Written Assessment

Task Description

In this assignment, you are required to develop, debug, and test a software application having a three-layered architecture with a front-end graphical user interface(GUI), middle layer implementing the business logic and a back-end to persist the data. You will also be documenting a test plan to systematically test the developed application and a quality plan retrospectively prepared to demonstrate how you would have commenced the application development with a quality plan in place to deliver a system meeting the quality requirements. You will be using the topics learned in weeks 1-10 in this assignment. Complete task details are in the Assignment 2 Specification document available from the Unit website.

Assessment Due Date

Week 11 Friday (29 May 2020) 11:45 pm AEST

Submit Assignment 2

Return Date to Students

Review/Exam Week Friday (12 June 2020)

Two weeks after submission

Weighting

40%

Assessment Criteria

This is an individual assignment and contributes to 40% of the total marks.

- Apply software design and development principles
- Complete appropriate documentation for requirements, system modelling, and design, software implementation, testing, and evolution.
- Illustrate a detailed design for a layered information system using appropriate design methods.
- Implement an information system, consisting of presentation, application and data layers.
- Use quality management in a range of different design and development scenarios.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Submission Instructions

Submit source code files (.java) and a document file (.doc or .docx) using the Assignment 2 submission link available from the unit website.

Learning Outcomes Assessed

- Apply software design and development principles
- Complete appropriate documentation for requirements, system modelling and design, software implementation, testing and evolution
- Illustrate a detailed design for a layered information system using appropriate design methods
- Implement an information system, consisting of presentation, application and data layers
- Use quality management in a range of different design and development scenarios.

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Information Technology Competence
- Ethical practice

Examination

Outline

Complete an invigilated examination.

Date

During the examination period at a CQUniversity examination centre.

Weighting

40%

Length

180 minutes

Exam Conditions

Open Book.

Materials

Dictionary - non-electronic, concise, direct translation only (dictionary must not contain any notes or comments).

Academic Integrity Statement

As a CQUniversity student you are expected to act honestly in all aspects of your academic work.

Any assessable work undertaken or submitted for review or assessment must be your own work. Assessable work is any type of work you do to meet the assessment requirements in the unit, including draft work submitted for review and feedback and final work to be assessed.

When you use the ideas, words or data of others in your assessment, you must thoroughly and clearly acknowledge the source of this information by using the correct referencing style for your unit. Using others' work without proper acknowledgement may be considered a form of intellectual dishonesty.

Participating honestly, respectfully, responsibly, and fairly in your university study ensures the CQUniversity qualification you earn will be valued as a true indication of your individual academic achievement and will continue to receive the respect and recognition it deserves.

As a student, you are responsible for reading and following CQUniversity's policies, including the [Student Academic Integrity Policy and Procedure](#). This policy sets out CQUniversity's expectations of you to act with integrity, examples of academic integrity breaches to avoid, the processes used to address alleged breaches of academic integrity, and potential penalties.

What is a breach of academic integrity?

A breach of academic integrity includes but is not limited to plagiarism, self-plagiarism, collusion, cheating, contract cheating, and academic misconduct. The Student Academic Integrity Policy and Procedure defines what these terms mean and gives examples.

Why is academic integrity important?

A breach of academic integrity may result in one or more penalties, including suspension or even expulsion from the University. It can also have negative implications for student visas and future enrolment at CQUniversity or elsewhere. Students who engage in contract cheating also risk being blackmailed by contract cheating services.

Where can I get assistance?

For academic advice and guidance, the [Academic Learning Centre \(ALC\)](#) can support you in becoming confident in completing assessments with integrity and of high standard.

What can you do to act with integrity?



Be Honest

If your assessment task is done by someone else, it would be dishonest of you to claim it as your own



Seek Help

If you are not sure about how to cite or reference in essays, reports etc, then seek help from your lecturer, the library or the Academic Learning Centre (ALC)



Produce Original Work

Originality comes from your ability to read widely, think critically, and apply your gained knowledge to address a question or problem